AUV Reactive Planning: Deepest Point

S. Pang
Department of Electrical Engineering
University of California, Riverside
Riverside, California
spang@ee.ucr.edu

J. A. Farrell
Department of Electrical Engineering
University of California, Riverside
Riverside, California
farrell@ee.ucr.edu

R. M. Arrieta
Ocean Technology Branch
SPAWAR Systems Center-San Diego
San Diego, California
arrieta@spawar.navy.mil

W. Li
Department of Computer Science
California State University
Bakersfield, California
wli@cs.csubak.edu

Abstract— **Autonomous Underwater Vehicles (AUV) capable of adapting their mission planning in response to sensor data have applicability in a variety of applications. For example, to find the source of a chemical in a fluid environment, an AUV could search a region until it detects the chemical plume, then steer the vehicle based on real-time chemical detections and flow information to maneuver the vehicle toward the odor source. Behavior based planning (BBP) is a reactive mission planning approach that is ideally suited for such applications. This article reviews behavior based planning, describes an example BBP designed to locate the deepest location within a test area, and includes results from in-water experiments on a REMUS AUV. This experiment was a test exercise in preparation for the use of BBP on a REMUS class vehicle to trace chemical plumes to their source.**

Keywords— **Autonomous vehicles, behavior based planning, gradient descent.**

I. Introduction

Various biological entities (e.g., moths, mosquitos, crabs, lobsters) are capable of tracking chemical plumes to the chemical source in turbulent fluid flow environments. Autonomous Underwater Vehicles (AUV) with such capabilities have utility in scientific and military applications. The ONR Chemical Sensing in the Marine Environment (CSME) and ONR/DARPA Chemical Plume Tracing (CPT) programs are intended to develop and demonstrate such capabilities. Efficient solution of the chemical plume tracing mission requires the AUV to have the ability to reactively change the mission plan (i.e., vehicle trajectory) in response to sensor information (e.g. chemical detection). Preliminary plume tracing test results from the CSME program are described in a companion paper [1]. This article focuses on preliminary testing of a reactive planning methodology known as behavior based planning (BBP).

On-vehicle in-water testing was accomplished on a REMUS vehicle. The REMUS was modified to contain a second PC-104 computer to implement the adaptive mission planning (AMP) algorithms. The AMP computer is connected to the main REMUS computer via serial port. When enabled, the AMP has access to REMUS state and sensor data and can issue altitude/depth, speed, and heading commands to the REMUS control system. The on-vehicle test and demonstration phase of the CSME program was split into three steps: (1) AMP enables and executes simple heading, speed, and depth commands; (2) AMP enables and executes a reactive mission; (3) AMP enables and executes CPT algorithms. Step 1 was completed in August 2002. The goal of Step 1 was to test the communication protocols between the main REMUS CPU and AMP. The objective of Step 2 was to debug and test the ability of AMP to reactively control the REMUS for completion of a mission. Due to the time and expense related to creating surrogate chemical plumes, we chose an artificial mission goal that was inexpensive, amenable to solution by a BBP approach, and required extensive reactive control of the vehicle for its solution. The test mission that we focused on for Step 2 was finding the deepest location within a specified operating area. The behavior based approach, algorithm details, and in-water results are described in the subsequent sections.

0-933957-31-9

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **01 SEP 2003**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **AUV Reactive Planning: Deepest Point**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Department of Electrical Engineering University of California, Riverside Riverside, California**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002146. Oceans 2003 MTS/IEEE Conference, Held in San Diego, California on September 22-26, 2003. U.S. Government or Federal Purpose Rights License, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**5**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

II. Behavior Based Planning

In the late 1970's and early 80's, Michael Arbib began to investigate models of animal intelligence from the biological and cognitive sciences in the hopes of gaining insight into what was missing in robotics [3]. At nearly the same time, Valentine Braitenberg studied how machine intelligence could be evolved by using sensor-motor pairs to design vehicle systems [4]. Later, a new generation of AI researchers began exploring the biological sciences in search of new organizing principles and methods of obtaining intelligence in robotic systems. They proposed the reactive behavior-based approach to designing robotic systems. A behavior is a mapping of sensory inputs to a pattern of motor actions, which then are used to achieve a task and will serves as the fundamental component for designing intelligent systems.

Rodney Brooks' subsumption architecture is the most influential of the purely reactive paradigms. Its basic idea is to describe a complex task by several behaviors with simple features [5]. Behavior-based control, however, has two significant disadvantages: Firstly, it may be difficult to formulate reactive behaviors quantitatively. Secondly, there might be no obvious approach to coordinating conflicts and competition among different reactive behaviors to achieve good performance. A usual approach to implementing behavior-based control is the use of artificial potential fields [6]. A drawback to this approach is that proper coordination of the reactive behaviors requires significant effort during the design phase to test and adjust potential fields parameters. This is especially true when these thresholds depend heavily on environmental parameters. Another alternative is to use artificial neural networks (ANNs). In the ANN approach, however, there are many "circumstance patterns" to be trained. Dangerous driving situations may occur when the robot meets circumstance patterns that have not yet been learned [10]. In [9], [7], [8], fuzzy-logic-based reactive behavior strategies are proposed to improve the performance of robot systems in two ways: First, reactive behavior is formulated by fuzzy sets and a rule base, and second, conflicts and competition among different types of reactive behavior are coordinated by fuzzy reasoning.

The application of BBP to the deepest point search problem is described in the reminder of this article. The behaviors described in the following sections used traditional logic as the basis for switching between the behaviors. In addition, there is one additional behavior referred to as the Cage behavior. The purpose of the Cage is to ensure that the vehicle does not leave the specified operating area (OpArea). The Cage function commands a heading that will steer the vehicle towards the interior of the OpArea. Switching of the Cage with the behaviors described below is based on simple fuzzy logic. If the vehicle is outside the OpArea, then the Cage heading is commanded. If the vehicle is inside the OpArea, near the boundary, and headed out, then a weighted average of the Cage and Deepest Point headings are issued. As the vehicle gets closer to the boundary, the Cage heading is weighted more heavily. This effectively removes the outward component from the vehicle velocity. When the vehicle is inside the OpArea and sufficiently far from the boundary, then the weighting of the Cage heading is zero.

III. Deepest Point Planner Overview

Finding the deepest location within a region of the sea with unknown bathymetry was selected as a test mission for reactive on-line planning. The problem will be solved using a REMUS class vehicle and BBP. Success in this mission is a step along the path towards solving the plume tracing problem.

The deepest point problem could be solved by performing a ladder search that covers the entire test region. For a rectangular test region with length L and width W, a lawn-mower search parallel to the region length with spacing d_L between search legs, the search path length is $p = L \times \frac{W}{d_L}$. Therefore, the whole search time is $t = \frac{p}{v} = \frac{L}{v}\frac{W}{d_L}$, where v is the speed of the vehicle and the time between legs has been neglected.

Since the goal is to find the deepest point and the bottom is continuous, the vehicle search efficiency could be improved by concentrating on deep regions and neglecting shallow regions. A BBP has been designed to implement and test this idea. The BBP will develop a function that locally approximates the bathymetry in the vicinity of the vehicle. This local map will be used to determine the direction of travel that maximizes the rate of depth increase. For this approach to succeed, the local map must be kept accurate and enough of the region must be explored to ensure that the global maximum depth (within the OpArea) is found. In the BBP approach, we break the problem down into simpler and more well defined tasks (Behaviors). Algorithms are designed to implement each of these behaviors and a switching mechanism is designed to select the most appropriate behavior

to control the vehicle at each time. For the deepest point problem we required three behaviors: GoTo, CurveFit, GradientFollow.

The GoTo behavior issues the guidance (i.e., heading, speed, and depth) commands required to maneuver the vehicle to a specified location. The CurveFit behavior maneuvers the vehicle appropriately to ensure that the solution for the local depth model parameters is well-conditioned. The GradientFollow behavior commands the heading corresponding to the gradient of the local bathymetric fit.

A behavior switching diagram is shown in Fig. 1. The switching logic is as follows:

Goto → CurveFit. When the vehicle is within distance R of the specified point, the BBP switches to CurveFit.

CurveFit → GradientFollow. When the vehicle completes a maneuver designed to yield a well-conditioned matrix inversion in the curve fit procedure, the curve fit parameters are computed. If the curve fit has no local maximum or if that maximum is sufficiently far from both the vehicle location and all previous points on a list of verified local maxima, then the BBP switches to GradientFollowing.

CurveFit → GoTo. When the curve fit has a local maximum and that maximum is sufficiently near either the vehicle location or a previous verified local maximum, then the vehicle has achieved and verified the local maximum of the current curve fit. The local maximum is added to the list of verified local maxima. Then, the BBP switches to GoTo.

GradientFollow → CurveFit. If the vehicle is following the gradient and the water depth is decreasing, then either the vehicle has past a local maximum or the curve fit is wrong. If the vehicle location is not near any point on the list of verified local maxima, then the BBP switches to curve fitting.

GradientFollow → GoTo. If the vehicle is following the gradient and the water depth is decreasing, then either the vehicle has past a local maximum or the curve fit is wrong. If the vehicle location is near a point on the list of verified local maxima, then the gradient has lead the vehicle to the basin of attraction of a previously known local maximum; therefore, the BBP switches to curve fitting.

This cycle repeats by going to p start locations, which have been selected to ensure sufficient coverage of the operating area. For the results shown herein, $p = 4$ and the four points are the corners of the operating area. At the completion of the p cycles, the point on the list of verified local maxima that has the largest depth is declared and the deepest point.

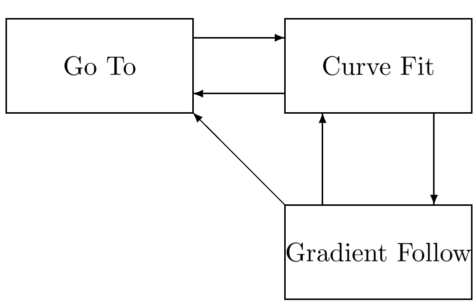

Fig. 1. Behavior Switching Diagram

A. Behavior: GoTo

At the start of the mission, the AMP has a list of p locations $(x_{s_i}, y_{s_i}),\ i \in [1,p]$ from which it will initiate gradient descent search. The GoTo behavior is used to maneuver the vehicle to each of these p locations to initialize gradient descent search. Each gradient descent will end at a local maxima within the OpArea. We keep a list of the local maxima as $(\bar{x}_i, \bar{y}_i),\ i \in [1,n]$. Therefore, at the end of p gradient descent trials we will have a list of $n \leq p$ local maxima. The greatest of these is declared as the deepest point. Then the mission concludes.

When the on-line planner is in GoTo, the output of the planner is the heading command

$$\psi_c = \arctan\left(y(t) - y_{s_i}, x(t) - x_{s_i}\right) \tag{1}$$

where $(x(t), y(t))$ is the current vehicle position and (x_{s_i}, y_{s_i}) is the present destination location. When the vehicle is within a small region of the destination location,

$$\sqrt{(x(t) - x_{s_i})^2 + (y(t) - y_{s_i})^2} < R \tag{2}$$

the planner switches from GoTo to CurveFit.

B. Behavior: CurveFit

The depth of the sea d is a function of position (x, y):

$$d = f(x, y) \tag{3}$$

Since this function is not known, we will use on-line data to estimate a second order Taylor series approximation to the depth data:

$$\begin{aligned}\hat{d}(x, y; x_0, y_0) \quad &= \quad a_1 + a_2 dx + a_3 dy + a_4 (dx)^2 \\ &\quad\; + a_5 (dx)(dy) + a_6 (dy)^2 \end{aligned} \tag{4}$$

where $dx = (x - x_0)$ and $dy = (y - y_0)$. This model is locally accurate in the vicinity of (x_0, y_0). The second

order a approximation $\hat{d}(x, y; x_0, y_0)$ will provide a gradient direction for maximal increase of $\hat{d}$ and an estimate of the location of a local maxima.

For convenience of notation, define

$$\hat{d}(x, y; x_0, y_0) = \mathbf{h}(x, y; x_0, y_0)\mathbf{a} \tag{5}$$

where

$$\begin{aligned} \mathbf{h} &= [1, dx, dy, dx^2, (dx)(dy), dy^2] \\ \mathbf{a} &= [a_1,\ a_2,\ a_3,\ a_4,\ a_5,\ a_6]'. \end{aligned}$$

The local model is specified by values of (x_0, y_0) and $\mathbf{h}$. These parameters will be computed from a set of data $\{(x_i, y_i, d_i)\}_{i=1}^N$ where $\{(x_i, y_i)\}_{i=1}^N$ are within a small region of interest near the current vehicle location and d_i represents the measured water depth at vehicle location (x_i, y_i).

The linearization point is selected as the center of the available dataset

$$x_0 = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad y_0 = \frac{1}{N}\sum_{i=1}^{N} y_i. \tag{6}$$

Equation (5) is a linear in the parameters $\mathbf{a}$. Therefore, the available set of data can be organized into a matrix equation. Let $dx_1 = (x_i - x_0)$, $dy_i = (y_i - y_0)$, $\mathbf{d} = [d_1,\ d_2,\ \cdots\ d_N]'$ and $\mathbf{h}_i = [1, dx_i, dy_i, dx_i^2, (dx_i)(dy_i), dy_i^2]$. Then, with the definition

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 \\ \vdots \\ \mathbf{h}_N \end{bmatrix} \tag{7}$$

we have

$$\mathbf{d} = \mathbf{H}\mathbf{a} \tag{8}$$

where $\mathbf{a}$ is the only unknown. The parameter vector $\mathbf{a}$ can be estimated as

$$\mathbf{a} = (\mathbf{H}'\mathbf{H})^{-1}\mathbf{H}'\mathbf{d}. \tag{9}$$

Reliable solution of eqn. (8) requires that the matrix $\mathbf{H}'\mathbf{H}$ be well-conditioned. The condition number of matrix $\mathbf{H}'\mathbf{H}$ depends on variation of the rows of $\mathbf{H}$, which is determined by the sensed positions (x_i, y_i) $i \in [1, N]$. If the set of points in $\{(x_i, y_i)\}_{i=1}^N$ are along a line, then the condition number will be large and matrix $\mathbf{H}'\mathbf{H}$ is ill-conditioned.

For this set of experiments, $N = 20$ and data (x_i, y_i, d_i) is saved at a 1 Hz rate. Upon switching to the CurveFit behavior, the heading command is held constant for $\frac{N}{2} = 10$ seconds. Then the heading command is changed by 90 deg. and held constant for another $\frac{N}{2} = 10$ seconds. This right angle trajectory ensures that the matrix $\mathbf{H}'\mathbf{H}$ is sufficiently well conditioned. At the end of N seconds, the CurveFit uses eqn. (9) to estimate $\mathbf{a}$. The extreme point of eqn. (8), if it exists, is at

$$dx^* = \frac{a_3 a_5 - 2a_2 a_6}{4a_4 a_6 - a_5^2} \qquad dy^* = \frac{a_2 a_5 - 2a_3 a_4}{4a_4 a_6 - a_5^2}. \tag{10}$$

This point is a local maximum if $a_4 < 0$ and $a_6 < 0$. If a local maximum exists, the location of the local maximum is $(x^*, y^*) = (x_0 + dx^*, y_0 + dy^*)$.

Now we compare the curve fit local maximum (x^*, y^*) with the list of previously verified local maxima $(\bar{x}_i, \bar{y}_i)$, $i \in [1, n]$, the current vehicle location, and the deepest point encountered thus far (x_d, y_d). If any of these match to accuracy R, then we assume we have found a new local maxima. If the local maximum is new, we store add it to the list and switch to GoTo. If there is no match, the BBP switches to GradientFollow.

C. Behavior: GradientFollow

After completion of the CurveFit behavior, we have a locally accurate estimated depth function $\hat{d}(x, y)$ and an estimated local maximum (x^*, y^*). The GradientFollow behavior is designed to verify the local maximum depth position. Since the gradient is the direction in which depth increases the fastest, the heading is commanded in the direction that causes gradient following. From equation (4), the gradient (G_x, G_y) at position (x, y) is

$$G_x = \frac{\partial F}{\partial x} = a_2 + 2a_4(x - x_0) + a_5(y - y_0) \tag{11}$$

$$G_y = \frac{\partial F}{\partial y} = a_3 + 2a_6(y - y_0) + a_5(x - x_0) \tag{12}$$

The corresponding heading command is

$$\psi_c = arctan(G_y, G_x). \tag{13}$$

While gradient following, the depth should be increased. If the depth does not increase over a T second interval, then the current curve fit is no longer accurate and the BBP switches back to the CurveFit behavior. If the vehicle successfully arrives near (x^*, y^*), then the estimated local deepest point has been verified. That point is added to the list of local maxima and the BBP switches to the GoTo behavior.

IV. In-water Results

These algorithms were tested on a REMUS class vehicle in South San Diego Bay in August 2002. Two runs

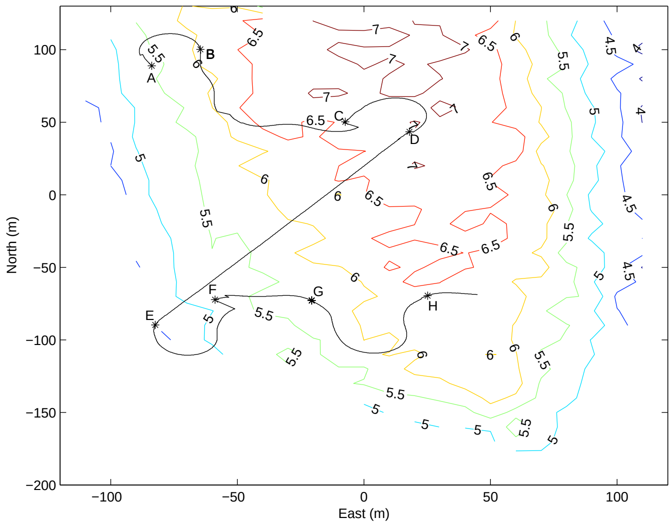

Fig. 2. A portion of the vehicle trajectory (solid line going through points A, B, ..., H) overlayed onto a post-mission reconstruction of the bottom contour map (colored lines with numbers indicating depth in meters).

were completed over the same operational area. The deepest points declared on the two runs were (32:37.9273N, 117:07.1744W) and (32:37.9206N, 117:07.1787W) which match to within 13 m.

Figure 2 displays a portion of the vehicle trajectory from the second experiment that was completed. Prior to this segment of the mission, AMP has just directed the vehicle to GoTo point A. The vehicle is performing a CurveFit between points A and B. AMP performs GradientFollowing of the curve fit between points B and C. Between C and D, AMP is performing a new CurveFit. This curve fit satisfies the stopping conditions, so AMP transitions to the next staring point indicated by E. Between E and F, AMP is performing a new CurveFit. AMP is GradientFollowing between F and G. Between G and H, AMP is CurveFitting.

V. Conclusion

This paper has reviewed the ideas of behavior based planning and described an application intended to implement and test behavior based planning on a REMUS class vehicle. The selected application was finding the deepest point within an operational area using gradient following and local curve fitting. The main test results indicate (1) that the AMP algorithms were computable within the computational constraints of the vehicle, (2) that AMP was able to effectively communicate with the REMUS control computer to achieve the trajectories desired for solving the problem, and (3) that the AMP algorithms succeeded in accurately finding the deepest point of the region.

This deepest point application was not of interest in its own right. Our main interest is in solving problems related to Chemical Plume Tracing. Due to the cost of creating chemical plumes of significant length, the deepest point application was selected as a lower cost test problem to ensure that the BBP algorithms were implementable on the vehicle and that the vehicle was controllable by AMP prior to actual tests related to chemical plume tracing.

Acknowledgment

This research was financially supported by the Office of Naval Research under grant N00014-01-1-0906. The authors appreciate the experimental and development support of the following people, without whom the in-water experiments could not have occurred: Rich Arrieta, Vladimir Djapic, Andrew Drieling, Brian Granger, Bruce Parks, Gerry Hong, Greg Packard, and Roger Stockey.

References

[1] J. A. Farrell, W. Li, S. Pang, and R. M. Arrieta, "Chemical Plume Tracing Experimental Results with a REMUS AUV," Proceeding OCEANS 2003.

[2] R. M. Arrieta, J. A. Farrell, W. Li, S. Pang,"Initial Development and Testing of an Adaptive Mission Planner for a Small Unmanned Underwater Vehicle" Proc. Of the 22nd Int. Conf. On Offshore Mechanics and Artic Engineering, OMEA2003-37273, AMSE.

[3] Arbib, M.,"Perceptional Structures and Distributed Motor Control," *Handbook of Physiology - The Nervous Systems II*, ed. Brooks, 1981.

[4] Braintenberg, V., *Vehicles: Experiments in Synthetic Psychology*, MIT Press, 1984.

[5] Brooks R. A. "A robust layered control system for a mobile robot," IEEE J. Robot. Automation RA-2, 14-23 (1986).

[6] Arkin R. C. and Murphy R. R. "Autonomous navigation in a manufacturing environment," IEEE Trans. Robot. Automation RA-6, 445-454 (1990).

[7] Li W. "Fuzzy-Logic-Based Reactive Behavior Control of an Autonomous Mobile System in unknown environments," Engineering Application of Artificial Intelligence, Vol. 7, No. 5, 521-531, 1994.

[8] Li W., Ma C.Y., and Wahl F.M., "A Neuro-Fuzzy System Architecture for Behavior-Based Control of a Mobile Robot in Unknown Environments," Fuzzy Sets and Systems, Vol. 87, 133-140, 1997.

[9] Saffiotti, A, Ruspini E.H., and Konolige K., "Blending Reactivity and goal-directness in a fuzzy controller," Proceedings of the IEEE International Conference on Fuzzy Systems, pp.134-139, 1993.

[10] Berns K., Dillmann R. and Hofsteller R. "An Application of a Backpropagation Network for the Control of a Tracking Behavior," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2426-2431, 1991.